

What Is Claimed Is:

- 1 1. A method for scheduling processes within an operating system
2 based upon virtual server identifiers, wherein the operating system supports
3 multiple virtual servers that operate within separate virtual environments on a
4 single computing platform, the method comprising:
5 detecting an event that causes a scheduling priority for a process to be
6 updated;
7 looking up a virtual server identifier for the process, wherein the virtual
8 server identifier specifies a virtual server and an associated virtual environment
9 that the process operates within;
10 using the virtual server identifier to look up a scheduling priority
11 associated with the virtual server; and
12 calculating an updated scheduling priority for the process based upon the
13 scheduling priority associated with the virtual server.
- 1 2. The method of claim 1, wherein calculating the updated scheduling
2 priority involves calculating the updated scheduling priority based upon:
3 a value, E , stored within a priority-related timer that keeps track of
4 execution time for the process;
5 a system priority, S_P , associated with the process; and
6 the scheduling priority, M , associated with the virtual server.
- 1 3. The method of claim 2, wherein calculating the updated scheduling
2 priority, P , involves calculating $P = S_P + S(E/M)$, wherein S is a tunable constant
3 value.

1 4. The method of claim 1, wherein the method further comprises:
2 receiving a command to adjust the scheduling priority associated with the
3 virtual server;

4 if the command is received from an authorized entity, adjusting the
5 scheduling priority associated with the virtual server so that the scheduling
6 priorities of all processes associated with the virtual server are modified.

1 5. The method of claim 1, wherein the method further comprises
2 charging a fee for hosting the virtual server, wherein the fee is based upon the
3 scheduling priority associated with the virtual server.

1 6. The method of claim 1, wherein detecting the event that causes the
2 scheduling priority for the process to be updated involves detecting one of:
3 the process entering a sleep state;
4 the process waking up from the sleep state; and
5 a priority-related timer associated with the process reaching a maximum
6 value.

1 7. The method of claim 1, wherein looking up the virtual server
2 identifier for the process involves looking up the virtual server identifier within a
3 process structure maintained by the operating system for the process.

1 8. A computer-readable storage medium storing instructions that
2 when executed by a computer cause the computer to perform a method for
3 scheduling processes within an operating system based upon virtual server
4 identifiers, wherein the operating system supports multiple virtual servers that

5 operate within separate virtual environments on a single computing platform, the
6 method comprising:
7 detecting an event that causes a scheduling priority for a process to be
8 updated;
9 looking up a virtual server identifier for the process, wherein the virtual
10 server identifier specifies a virtual server and an associated virtual environment
11 that the process operates within;
12 using the virtual server identifier to look up a scheduling priority
13 associated with the virtual server; and
14 calculating an updated scheduling priority for the process based upon the
15 scheduling priority associated with the virtual server.

1 9. The computer-readable storage medium of claim 8, wherein
2 calculating the updated scheduling priority involves calculating the updated
3 scheduling priority based upon:
4 a value, E , stored within a priority-related timer that keeps track of
5 execution time for the process;
6 a system priority, S_P , associated with the process; and
7 the scheduling priority, M , associated with the virtual server.

1 10. The computer-readable storage medium of claim 9, wherein
2 calculating the updated scheduling priority, P , involves calculating $P = S_P +$
3 $S(E/M)$, wherein S is a tunable constant value.

1 11. The computer-readable storage medium of claim 8, wherein the
2 method further comprises:

1 receiving a command to adjust the scheduling priority associated with the
2 virtual server;
3 if the command is received from an authorized entity, adjusting the
4 scheduling priority associated with the virtual server so that the scheduling
5 priorities of all processes associated with the virtual server are modified.

1 12. The computer-readable storage medium of claim 8, wherein the
2 method further comprises charging a fee for hosting the virtual server, wherein the
3 fee is based upon the scheduling priority associated with the virtual server.

1 13. The computer-readable storage medium of claim 8, wherein
2 detecting the event that causes the scheduling priority for the process to be
3 updated involves detecting one of:
4 the process entering a sleep state;
5 the process waking up from the sleep state; and
6 a priority-related timer associated with the process reaching a maximum
7 value.

1 14. The computer-readable storage medium of claim 8, wherein
2 looking up the virtual server identifier for the process involves looking up the
3 virtual server identifier within a process structure maintained by the operating
4 system for the process.

1 15. An apparatus that schedules processes within an operating system
2 based upon virtual server identifiers, wherein the operating system supports
3 multiple virtual servers that operate within separate virtual environments on a
4 single computing platform, the apparatus comprising:

5 a detection mechanism that is configured to detect an event that causes a
6 scheduling priority for a process to be updated;
7 a lookup mechanism that is configured to look up a virtual server identifier
8 for the process, wherein the virtual server identifier specifies a virtual server and
9 an associated virtual environment that the process operates within;
10 wherein the lookup mechanism is additionally configured to use the virtual
11 server identifier to look up a scheduling priority associated with the virtual server;
12 and
13 a calculating mechanism that is configured to calculate an updated
14 scheduling priority for the process based upon the scheduling priority associated
15 with the virtual server.

1 16. The apparatus of claim 15, wherein the calculating mechanism is
2 configured to calculate the updated scheduling priority based upon:
3 a value, E , stored within a priority-related timer that keeps track of
4 execution time for the process;
5 a system priority, S_P , associated with the process; and
6 the scheduling priority, M , associated with the virtual server.

1 17. The apparatus of claim 16, wherein the calculating mechanism is
2 configured to calculate the updated scheduling priority, P , by calculating
3 $P = S_P + S(E/M)$, wherein S is a tunable constant value.

1 18. The apparatus of claim 15, further comprising a priority adjustment
2 mechanism that is configured to:
3 receive a command to adjust the scheduling priority associated with the
4 virtual server; and to

1 adjust the scheduling priority associated with the virtual server so that the
2 scheduling priorities of all processes associated with the virtual server are
3 modified, if the command is received from an authorized entity.

1 19. The apparatus of claim 15, further comprising a fee calculation
2 mechanism that is configured to calculate a fee for hosting the virtual server based
3 upon the scheduling priority associated with the virtual server.

1 20. The apparatus of claim 15, wherein the detection mechanism is
2 configured to detect one of:
3 the process entering a sleep state;
4 the process waking up from the sleep state; and
5 a priority-related timer associated with the process reaching a maximum
6 value.

1 21. The apparatus of claim 15, wherein the lookup mechanism is
2 configured to look up the virtual server identifier for the process by looking up the
3 virtual server identifier within a process structure maintained by the operating
4 system for the process.